

A Design Of A Low Power Delay Buffer Using Ring Counter Addressing Schemes

B.R.B Jaswanth

St. Theresa Institute of Engineering and Technology
Gudiwada, India

R.V.S Rayudu, K.Mani babu, R.Himaja, L.Veda kumar

V.K.R, V.N.B & A.G.K College of Engineering
Gudiwada, India

Abstract— This work presents circuit design of a low-power delay buffer. The proposed delay buffer uses several new techniques to reduce its power consumption. Since delay buffers are accessed sequentially, it adopts a ring-counter addressing scheme. In the ring counter, double-edge-triggered (DET) flip-flops are utilized to reduce the operating frequency by half and the C-element gated-clock strategy is proposed. A novel gated-clock-driver tree is then applied to further reduce the activity along the clock distribution network. Moreover, the gated-driver-tree idea is also employed in the input and output ports of the memory block to decrease their loading, thus saving even more power.

Keywords- Low Power Delay Buffer, Double Edge Triggered Flip Flop, Ring Counter.

I. INTRODUCTION

The skyrocketing increasing transistor count and circuit density of modern very large scale integrated (VLSI) circuits have made them extremely difficult and expensive to test comprehensively. In a digital processing chip of mobile communications, the delay buffer takes up a large portion of the circuit layout. If the power consumption of the delay buffer could be reduced significantly, the overall power consumption of the digital processing chip could be reduced significantly as well. On the other hand, as these chips are working at even higher operation frequencies, a new, low-power delay buffer should be operable under high frequencies. FIG. 1 is a schematic diagram showing a conventional delay buffer having a length N and data width W bits using shift registers. As illustrated, the delay buffer contains N times. W shift registers 10, arranged between the input and the output in N stages, each with W shift registers. The N times W shift registers are triggered by a same clock signal CLK . For every clock period of CLK , W -bit data is shifted from W shift registers of a previous stage to those of a next stage, and so on. A W -bit data input N clock periods ago therefore would be delayed and output after N clock periods. The clock signal CLK is provided to all N times W shift registers, contributing to the high power consumption. Moreover, the N times W shift registers would also take up a large die area. Therefore, in real life, delay buffer such as the one.

One of the common delay buffer implementation is a dual-port SRAM memory whose operation is different from that of the shift-register-based delay buffer. For an N times W SRAM-based delay buffer, there is no data movement between stages. Instead, at every clock period, a W -bit data is written to one of

the N times W storage locations of the SRAM-based delay buffer, and another W -bit data that is written N clock periods ago is output. The power consumption of a SRAM-based delay buffer is mainly from the address decoder and the drivers for its input and output ports. As memory related technology has already quite mature and satisfactory results in terms of layout area and speed are achievable. Therefore in reality a delay buffer is often implemented using SRAM memory.

II. MEMORY ORGANIZATION

Memory organization is two-fold. First we discuss the hardware (physical) organization, then the internal architecture. The type of computer and its size do not reflect the type of memories that the computer uses. Some computers have a mixture of memory types. For example, they may use some type of magnetic memory (core or film) and also a semiconductor memory (static or dynamic). They also have a read-only memory which is usually a part of the CPU. Memory in a computer can vary from one or more modules to one or more pcb's, depending on the computer type. The larger mainframe computers use the modular arrangement, multiple modules (four or more), to make up their memories whereas, minicomputers and microcomputers use chassis or assemblies, cages or racks, and motherboard or backplane arrangements. Minis and micros use multiple components on one PCB or groups of PCB have to form the memory.

There are several ways to organize memories with respect to the way they are connected to the cache:

- one-word-wide memory organization
- wide memory organization
- interleaved memory organization
- independent memory organization

III. DELAY BUFFER

This section describes PJMEDIA's implementation of delay buffer. Delay buffer works quite similarly like a fixed jitter buffer, that is it will delay the frame retrieval by some interval so that caller will get continuous frame from the buffer. This can be useful when the operations are not evenly interleaved, for example when caller performs burst of put() operations and then followed by burst of operations. With using this delay buffer, the buffer will put the burst frames into a buffer so that get() operations will always get a frame from the buffer (assuming that the number of get() and put() are matched).

The buffer is adaptive, that is it continuously learns the optimal delay to be applied to the audio flow at run-time. Once the optimal delay has been learned, the delay buffer will apply this delay to the audio flow, expanding or shrinking the audio samples as necessary when the actual audio samples in the buffer are too low or too high.

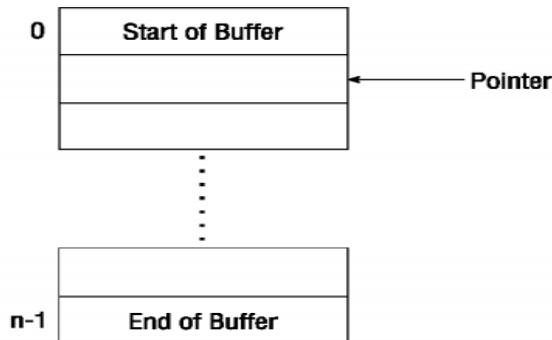


Figure 1. Memory Buffer

A. EXISTING TECHNIQUE

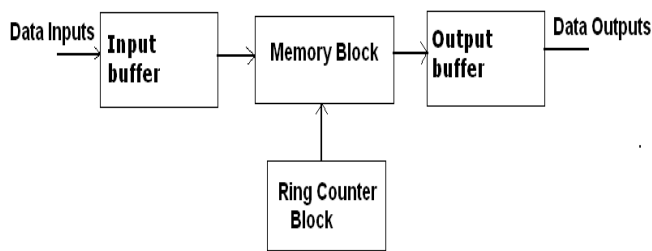


Figure 2. Block Diagram of Existing Technique

1) INPUT BUFFER

The Input buffer is also commonly known as the input area or input block. When referring to computer memory, the input buffer is a location that holds all incoming information before it continues to the CPU for processing.

Input buffer can be also used to describe various other hardware or software buffers used to store information before it is processed.

Some scanners (such as those which support “include” files) require reading from several input streams. As flex scanners do a large amount of buffering, one cannot control where the next input will be read from by simply writing a `YY_INPUT()` which is sensitive to the scanning context. `YY_INPUT()` is only called when the scanner reaches the end of its buffer, which may be a long time after scanning a statement such as an include statement which requires switching the input source.

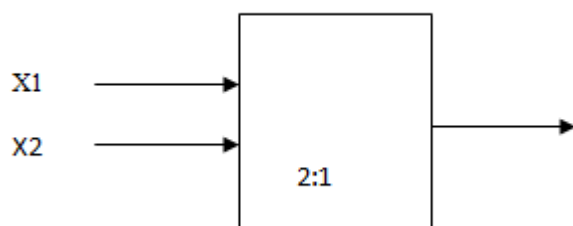


Figure 3. Input Buffer

2) MEMORY BLOCK

Random-access memory (RAM) is a form of computer data storage. Today, it takes the form of integrated circuits that allow stored data to be accessed in any order (that is, at random). "Random" refers to the idea that any piece of data can be returned in a constant time, regardless of its physical location and whether it is related to the previous piece of data. The word "RAM" is often associated with volatile types of memory (such as DRAM memory modules), where the information is lost after the power is switched off. Many other types of memory are RAM as well, including most types of ROM and a type of flash memory called NOR-Flash.

Scan design has been the backbone of design for testability (DFT) in industry for about three decades because scan-based design can successfully obtain controllability and observability for flip-flops. Serial Scan design has dominated the test architecture because it is convenient to build. However, the serial scan design causes unnecessary switching activity during testing which induce unnecessarily enormous power dissipation. The test time also increases dramatically with the continuously increasing number of flip-flops in large sequential circuits even using multiple scan chain architecture. An alternate to serial scan architecture is Random Access Scan (RAS). In RAS, flip-flops work as addressable memory elements in the test mode which is a similar fashion as random access memory (RAM). This approach reduces the time of setting and observing the flip-flop states but requires a large overhead both in gates and test pins. Despite of these drawbacks, the RAS was paid attention by many researchers in these years. This paper takes a view of recently published papers on RAS and rejuvenates the random access scan as a DFT method that simultaneously address three limitations of the traditional serial scan namely, test data volume, test application time, and test power.

3) RING COUNTER

A ring counter is a type of counter composed of a circular shift register. The output of the last shift register is fed to the input of the first register.

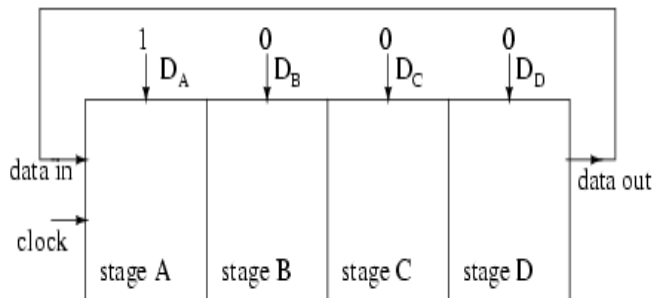
There are two types of ring counters:

- A straight ring counter or Overbeck counter connects the output of the last shift register to the first shift register input and circulates a single one (or zero) bit around the ring. For example, in a 4-register one-hot counter, with initial register values of 1000, the repeating pattern is: 1000, 0100, 0010, 0001, 1000... . Note that one of the registers must be pre-loaded with a 1 (or 0) in order to operate properly.
- A twisted ring counter (also called Johnson counter or Moebius counter) connects the complement of the output of the last shift register to its input and circulates a stream of ones followed by zeros around the ring. For example, in a 4-register counter, with initial register values of 0000, the repeating pattern is: 0000, 1000, 1100, 1110, 1111, 0111, 0011, 0000... .

If the output of a shift register is fed back to the input. a ring counter results. The data pattern contained within the shift register will re-circulate as long as clock pulses are applied. For example, the data pattern will repeat every four clock pulses in the figure below. However, we must load a data pattern. All 0's

or all 1's doesn't count. Is a continuous logic level from such a condition useful?

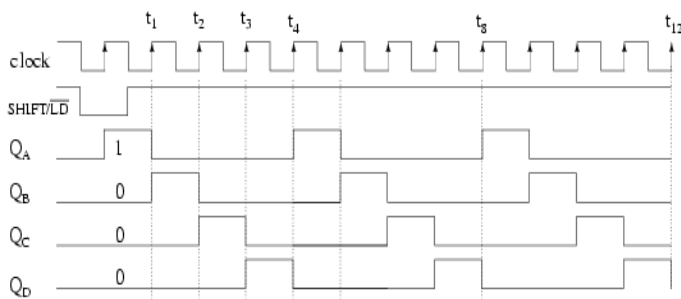
We make provisions for loading data into the parallel-in/serial-out shift register configured as a ring counter below. Any random pattern may be loaded. The most generally useful pattern is a single 1.



Parallel-in, serial-out shift register configured as a ring counter

Figure 4. Parallel-in Serial-Out Shift Register

Loading binary 1000 into the ring counter, above, prior to shifting yields a viewable pattern. The data pattern for a single stage repeats every four clock pulses in our 4-stage example. The waveforms for all four stages look the same, except for the one clock time delay from one stage to the next. See figure below.



Load 1000 into 4-stage ring counter and shift

Figure 5. 4-Stage Ring Counter and Shift

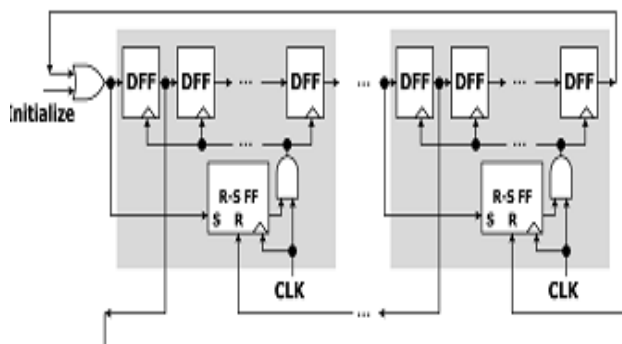


Figure 6. Ring counter with SR flip-flops

The above block diagram shows the power controlled Ring counter. First, total block is divided into two blocks. Each block is having one SR FLIPFLOP controller.

B. PROPOSED TECHNIQUE:

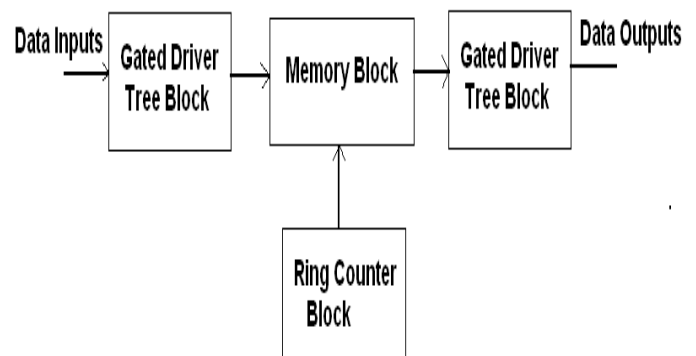


Figure 7. Block Diagram of Proposed Technique

1) GATED DRIVER TREE:

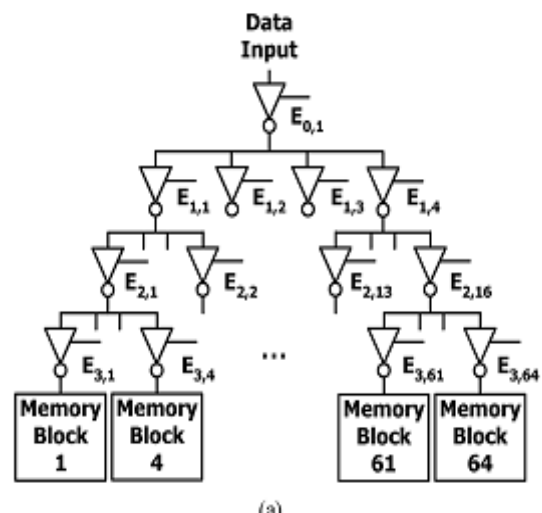


Figure 8. Gated Driver Tree

It is derived from the same clock gating signals of the blocks that they drive. Thus, in a quad-tree clock distribution network, the "gate" signal of the "th" gate driver at the "th" level (CKE) should be asserted when the active DET flip-flop .

2) MODIFIED RING COUNTER:

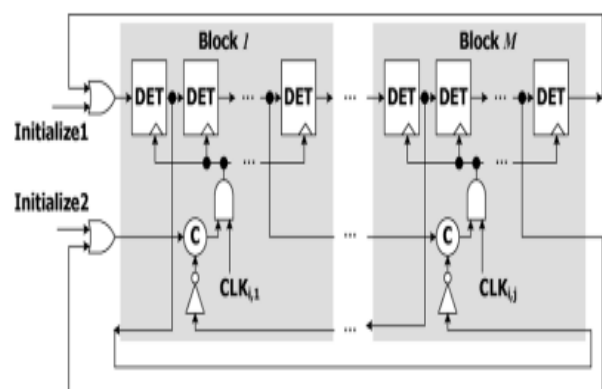


Figure 9. Modified Ring Counter

DET (Double edge triggered flip-flops: double-edge-triggered (DET) flip-flops are utilized to reduce the operating frequency by half. The logic construction of a double-edge-triggered (DET) flip-flop, which can receive input signal at two levels the clock, is analyzed and a new circuit design of CMOS DET. In this paper, we propose to use double-edge-triggered (DET) flip-flops instead of traditional DFFs in the ring counter to halve the operating clock frequency. Double edge-triggered flipflops are becoming a popular technique for low-power designs since they effectively enable a halving of the clock frequency. The paper by Hossain et al[1] showed that while a single-edge triggered flipflop can be implemented by two transparent latches in series, a double edge-triggered flipflop can be implemented by two transparent latches in parallel; the circuit in Fig. 1 was given for the static flipflop implementation. The clock signal is assumed to be inverted locally. In high noise or low-voltage environments, Hossain et al noted that the p-type pass-transistors may be replaced by n-types or that all pass-transistors may be replaced by transmission gates.

IV. SIMULATION AND SYNTHESIS

To obtain a further insight about the scalability of the proposed delay buffer architecture in nanometer CMOS technology, simulations of the proposed buffer with several different lengths in 90-nm and 65-nm CMOS technology are run. To alleviate the leakage power problem, dual-Vt MOS transistors are used in the 90-nm technology.

The low Vt MOS transistors only exist in write-enable and read-enable pass gates between bit lines and memory cells to provide enough drying, the supply voltages are 1V and 0.85V in 90-nm cases, respectively, and the operating frequency is 200 MHz. The total power consumption in normal operation mode is not logarithmically proportional to the length of the delay buffer in snorted. Instead, due to the quad tree structure for all the driving circuitry, delay buffers of length and have approximate dynamic power because basically these two cases activate the same number of drivers. One can see that the superiority of the proposed circuit is still obvious in 90-nm technology in that the leakage power is almost negligible.

A. SIMULATION AND SYNTHESIS POWER REPORTS OF THE EXISTING DELAY BUFFER

Even in the more advanced 65-nm technology, the leakage power can be controlled to within unacceptable level for medium-length delay buffers with the dual-Vt approach. For longer-length delay buffers and for more advanced technology, other leakage reduction techniques such as the “sleep” transistors in SRAM (Latch).

To highlight the result of the proposed delay buffer, the simulation of the previous existing buffer is of paramount importance. Hence the simulation and the synthesis results of the existing buffer are preceded by the simulation and synthesis power reports of the proposed delay buffer.

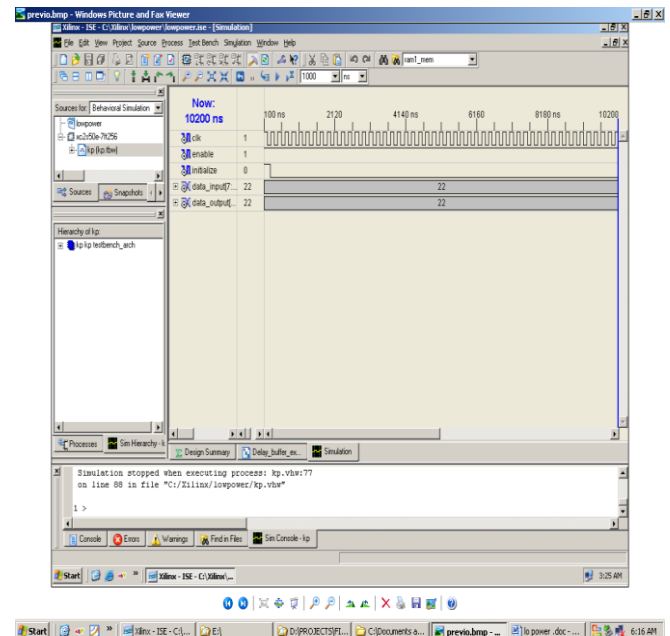


Figure 10. Simulation Power Report of the Existing Delay Buffer

The existing methodology is simulated using the synthesis and simulation tools and the analysis is drawn out as shown above which tells that the power levels and the time taken are significantly high and the relative comparison is done by simulating the present, proposed methodology.

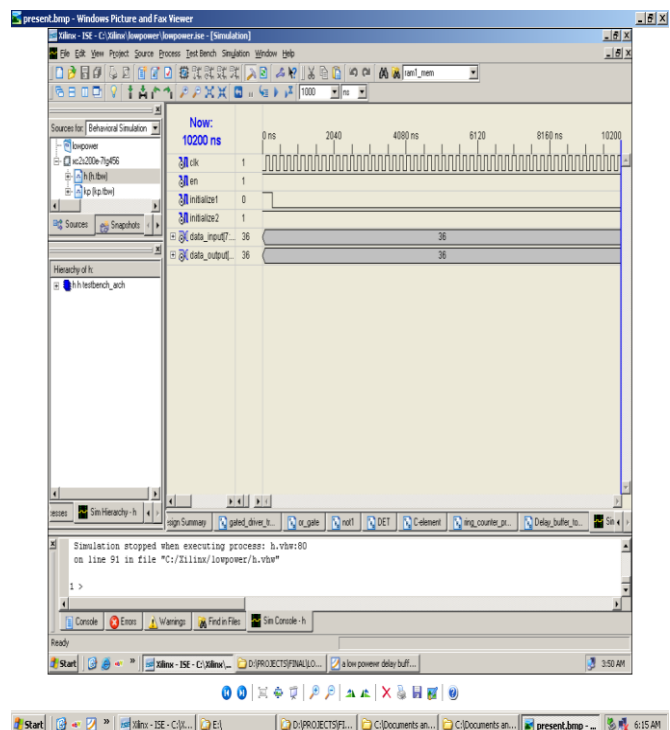


Figure 11. The Simulation Project Report of the Proposed Delay Buffer

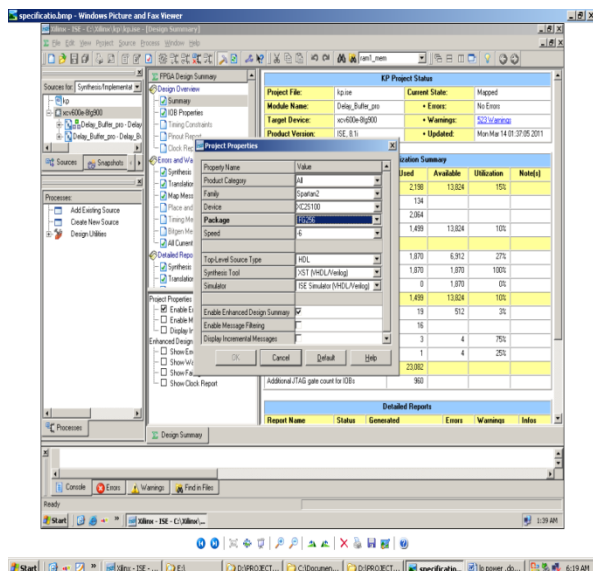


Figure 12. Synthesis Power Report of the Existing Delay Buffer

The properties and simulation results are analyzed by implementing on the tool simulator Xilinx with Spartan and device XC25100 with FG256 package. The synthesis tool is XST (verilog/VHDL) and the simulation tool is ISE simulator(VHDL/Verilog)

To highlight the result of the proposed delay buffer, the simulation of the previous existing buffer is of paramount importance. Hence the simulation and the synthesis results of the existing buffer are processed by the simulation and synthesis power reports of the proposed delay buffer.

The power report represents the consumption criteria and dissipation criteria of the existing and the proposed present techniques. The relative comparison is made based on the two methodologies using the same synthesis and simulation tools and the proportionate results speak that the robustness of the present proposed design methodology is in the lime light.

V. OVER-ALL POWER REPORT

A. EXISTING

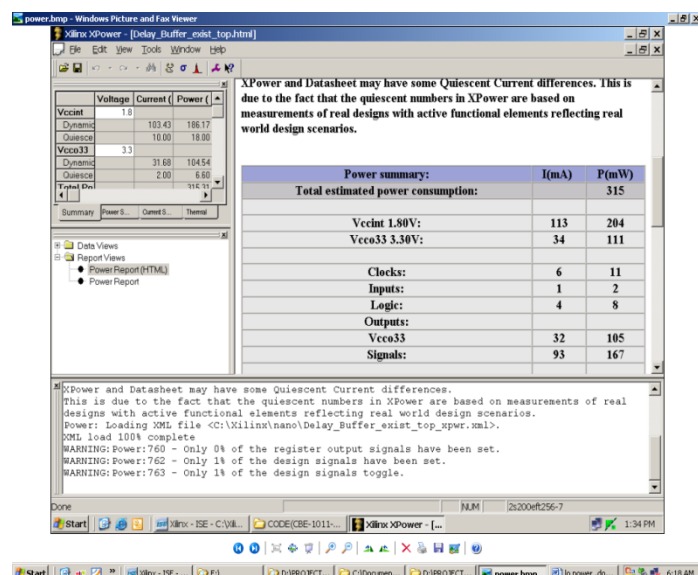


Figure 13. Existing Technique over All Power Report

B. PROPOSED

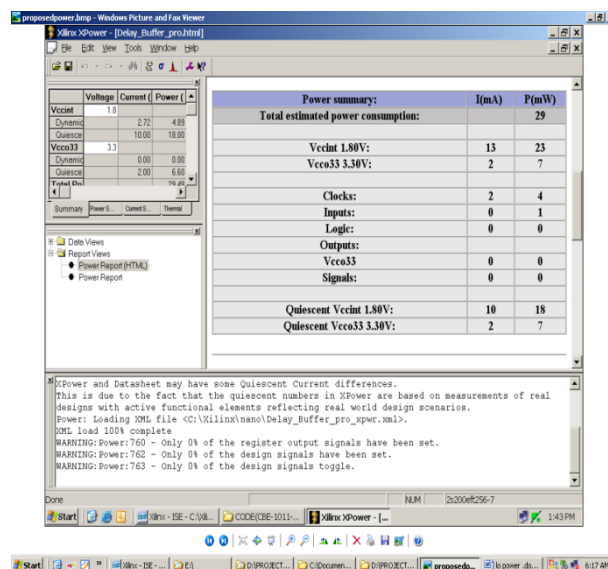


Figure 14. Proposed Technique over All Power Report

VI. CONCLUSION

We have analyzed the existing method and proposed method using the same synthesis and simulator tool. The simulation and the synthesis results of the existing buffer are processed by the simulation and synthesis power reports of the proposed delay buffer and the proportionate results speak that the robustness of the present proposed design methodology is in the lime light.

REFERENCES

- [1] W. Eberle *et al.*, "80-Mb/s QPSK and 72-Mb/s 64-QAM flexible and scalable digital OFDM transceiver ASICs for wireless local area Networks in the 5-GHz band," *IEEE J. Solid-State Circuits*, vol. 36, no. 11, pp. 1829–1838, Nov. 2001.
- [2] M. L. Liou, P. H. Lin, C. J. Jan, S. C. Lin, and T. D. Chiueh, "Design of an OFDM baseband receiver with space diversity," *IEE Proc. Commun.*, vol. 153, no. 6, pp. 894–900, Dec. 2006.
- [3] G. Pastuszak, "A high-performance architecture for embedded block coding in JPEG 2000," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 9, pp. 1182–1191, Sep. 2005.
- [4] W. Li and L. Wanhammar, "A pipeline FFT processor," in *Proc. Workshop Signal Process. Syst. Design Implement.*, 1999, pp. 654–662.
- [5] E. K. Tsem and T. H. Meng, "A low-power video-rate pyramid VQ decoder," *IEEE J. Solid-State Circuits*, vol. 31, no. 11, pp. 1789–1794, Nov. 1996.
- [6] N. Shibata, M. Watanabe, and Y. Tanabe, "A current-sensed high-speed and low-power first-in-first-out memory using a wordline/bitline-swapped dual-port SRAM cell," *IEEE J. Solid-State Circuits*, vol. 37, no. 6, pp. 735–750, Jun. 2002.
- [7] E. Sutherland, "Micropipelines," *Commun. ACM*, vol. 32, no. 6, pp. 720–738, Jun. 1989.
- [8] A Low-Power Delay Buffer Using Gated Driver Tree Po-chun hsieh, jing-siang jhuang, pei-yun tsai, *member, IEEE*, and tzi-dar chiueh, *senior member, IEEE*.